Familial monads and structural operational semantics

Tom Hirschowitz

Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA Chambéry, France

Journées inaugurales Logique, Homotopie, Catégories October 18, 2018

Structural operational semantics

- Notes by Plotkin (1981) : method rather than theory, by example.
- Describe dynamics of programming languages, syntactically.
 - Terms from algebraic signature.
 - Dynamics as a (labelled) transition system.
 - Basic idea : describe behaviour of each operation.

$$\frac{\dots \quad x_i \xrightarrow{a_i} y_i \quad \dots}{f(x_1, \dots, x_n) \xrightarrow{a} M(y_1, \dots, y_n)}$$

Structural : behaviour of system determined by its components.

Disturbing operation : bisimilarity in π not a congruence !

Structural \Rightarrow compositional.

Introduction			
Formats			

De Simone (1985) : rule format.

Algebraic signature + transition system specification

 \rightarrow transition system.

- Specification complies with format \implies transition system behaves well.
- E.g.,
 - (weak) bisimilarity is a congruence,
 - conservative extension,
 - bisimulation up to X is sound.

A wealth of formats

Since then, lots of different formats, combining :

- negative premises,
- predicates,
- look ahead.
- terms as labels.
- variable binding,...

Functorial operational semantics

- Attempt to tame the diversity of formats.
- Appealing simplicity :
 - terms = monad T.
 - labels = comonad L.
 - rules = distributive law $TL \rightarrow LT$.
- But not so widely adopted.
- Possible reasons :
 - too abstract.
 - not expressive enough (e.g., no negative premises afaik),
 - does not scale well to variable binding.
- Simplifying attempt by Staton (2008) :
 - SOS = monad on labeled relations.
 - Better treatment of variable binding.
 - But not better adopted.

Introduction			
6			
Proposa			

Two distinct goals :

- $1. \ \mbox{Find}$ the right language for describing
 - what goes on in proofs of congruence of bisimilarity, etc,...,
 - under which hypotheses.
- 2. Generate instances satisfying the hypotheses.

Here : focus on (1).

Abstract over the following.

- Bisimulation : by lifting (cf. presheaf models), in a "category of transition systems", *C*.
- SOS specifications : monad *T* on *C*.
 Morally : saturation by the given rules.
- Model of a SOS specification : \mathcal{T} -algebra.
- Congruence proof \leftarrow familiality of \mathcal{T} ,...

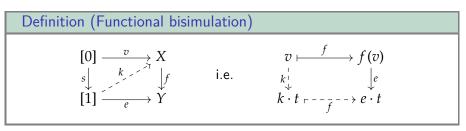
My first transition category

Categories that look like transition systems and simulations.

Baby example

(Directed, multi-)graphs, Gph.

- Untyped, one label.
- Presheaves over $s, t: [0] \rightrightarrows [1]$.



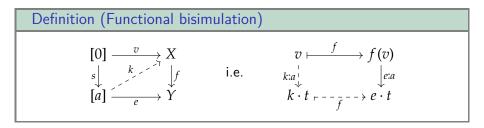
A transition category with basic labels

- Let A be the considered set of labels.
- Presheaves over Ω_A :



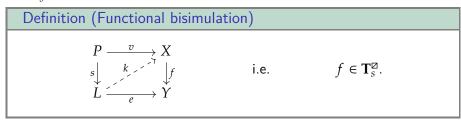
- Any $X \in \widehat{\Omega_A}$ has
 - a set of vertices X[0],
 - a set of *a*-transitions X[a] for all $a \in A$, each with its source and target.

A transition category with basic labels



	Transition categories				
Transitic	on categories				
Definit	ion				
Category with distinguished cospans					
		$P \xrightarrow{s} L \xleftarrow{t} Q$	2		
+ finite completeness, cocompleteness, well-poweredness, images, and tininess of all $P \in \mathbf{P}$.					

Let \mathbf{T}_s denote the set of all such $s: P \to L$.



SOS specifications as monads

Idea (Staton) : view SOS rules as endofunctors.

Example, on $\widehat{\Omega}_{A}$ SOS specification $S \rightsquigarrow \text{monad } \mathcal{T}_S$: • $\mathcal{T}_{S}(X)[0]$: terms with constants in X, • $\mathcal{T}_{S}(X)[a]$: derivations with transition axioms in X, • multiplication $\mathscr{T}_{S}^{2}(X)[a] \to \mathscr{T}_{S}(X)[a]$: plugging derivations. Example CCS.

 \rightarrow basic abstract framework : transition category with a monad on it.

Congruence of bisimilarity

Will follow from :

Theorem

If $f:R \to X$ is a functional bisimulation and X is a ${\mathcal T}\mbox{-algebra},$ then so is

$$\mathscr{T}(R) \xrightarrow{\mathscr{T}(f)} \mathscr{T}(X) \xrightarrow{a} X,$$

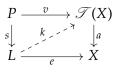
up to hypotheses.

Hypothesis 1 : compositionality

Generally a vague concept (thanks for asking!).

Definition

An algebra $a: \mathcal{T}(X) \to X$ is compositional iff it is a functional bisimulation.



Morally : any transition $C[x_1, ..., x_n] \xrightarrow{\alpha} x'$ decomposes as

$$\dots$$
 $x_i \xrightarrow{\alpha_i} y_i \dots$

$$C[x_1,...,x_n] \xrightarrow{\alpha} E[y_1,...,y_n]$$

Congruence of bisimilarity

Will follow from :

Theorem

If $f: R \to X$ is a functional bisimulation and X is a compositional \mathcal{T} -algebra, then so is

$$\mathscr{T}(R) \xrightarrow{\mathscr{T}(f)} \mathscr{T}(X) \xrightarrow{a} X,$$

up to hypotheses.

It now suffices to prove that $\mathcal{T}(f)$ is a functional bisimulation.

- Consider any $C[r_1, ..., r_n] \in \mathscr{T}(R)$ and let $x_i = f(r_i)$.
- Assume $C[x_1, ..., x_n] \xrightarrow{L} E[x'_1, ..., x'_m]$ (say m = n to simplify !).
- But f is a bisimulation, so find

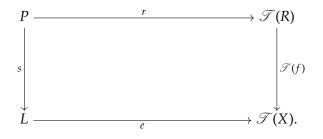
$$C[r_1, ..., r_n] \xrightarrow{\mathcal{T}(f)} C[x_1, ..., x_n] \xrightarrow{\downarrow E[e_1, ..., e_n]:L} D[x'_1, ..., x'_m].$$

- That's the intuition. In practice :
 - transition contexts E are not first-class citizens,
 - → induction on C.

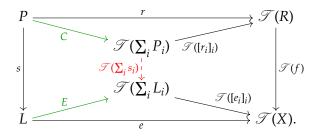
- Consider any $C[r_1, ..., r_n] \in \mathscr{T}(R)$ and let $x_i = f(r_i)$.
- Assume $C[x_1, ..., x_n] \xrightarrow{L} E[x'_1, ..., x'_m]$ (say m = n to simplify !).
- But f is a bisimulation, so find

- That's the intuition. In practice :
 - transition contexts E are not first-class citizens,
 - → induction on C.

In the abstract framework

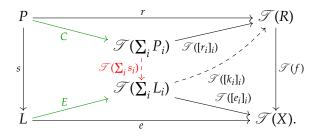


In the abstract framework

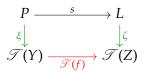


Familiality !

In the abstract framework



 \mathbf{T}_{s} -familiality



If ξ and ζ are generic, then $f \in \mathbb{Z}(\mathbf{T}_s^{\mathbb{Z}})$.

Standard definition of bisimulation up to context

R progresses to $\mathscr{C}(R)$, where

 $\mathcal{C}(R) \coloneqq \{(C[P_1, \dots, P_n], C[Q_1, \dots, Q_n]) \mid P_i \mid R \mid Q_i\}.$

$$\begin{array}{cccc} x & R & y \\ \downarrow & & \downarrow \\ x' & \mathscr{C}(R) & \exists y' \end{array}$$

Standard definition of bisimulation up to context

R progresses to $\mathscr{C}(R)$, where

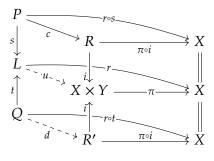
 $\mathcal{C}(R) \coloneqq \{ (C[P_1, \dots, P_n], C[Q_1, \dots, Q_n]) \mid P_i \mid R \mid Q_i \}.$

Generalises to R progresses to R'.

Progression in the abstract framework



- Relations $R, R' \hookrightarrow X \times Y$ in transition category.
- $R \sim R'$ iff



and symmetrically for Υ .

Up to

Example : bisimulation up to context

 $R \rightsquigarrow \mathscr{T}(R).$

		Up to	
But wait.			

Question

Does $R \rightarrow R$ iff R is a bisimulation?

- Not quite, but artefact of formalism.
- Reason : in $R \rightsquigarrow R$, $R \hookrightarrow X \times Y$ may have no transition.
- Good news, we can add them :

Proposition

Under mild hypotheses, factors as

$$R \to \overline{R} \to X \times Y$$

with \overline{R} a bisimulation.

Soundness of bisimulation up to context

Theorem

Under hypotheses, $R \rightsquigarrow \mathcal{T}(R)$ entails $\mathcal{T}(R) \rightsquigarrow \mathcal{T}(R)$.

Corollary

Any bisimulation up to context embeds into some bisimulation.

			Conclusion
Conclusi	ion		

Summary :

- SOS specification = monad on a transition category.
- Hypotheses ⇒
 - congruence of bisimilarity,
 - soundness of bisimulation up to context.

Perspectives :

- Existing formats → instances?
- More general format along the lines of free monads.
- Other up to techniques.
- Related questions, e.g., process equations, environmental bisimulation.
- Broader scope : analytic monads, to accomodate structural congruence.
- Go quantitative ?